

20

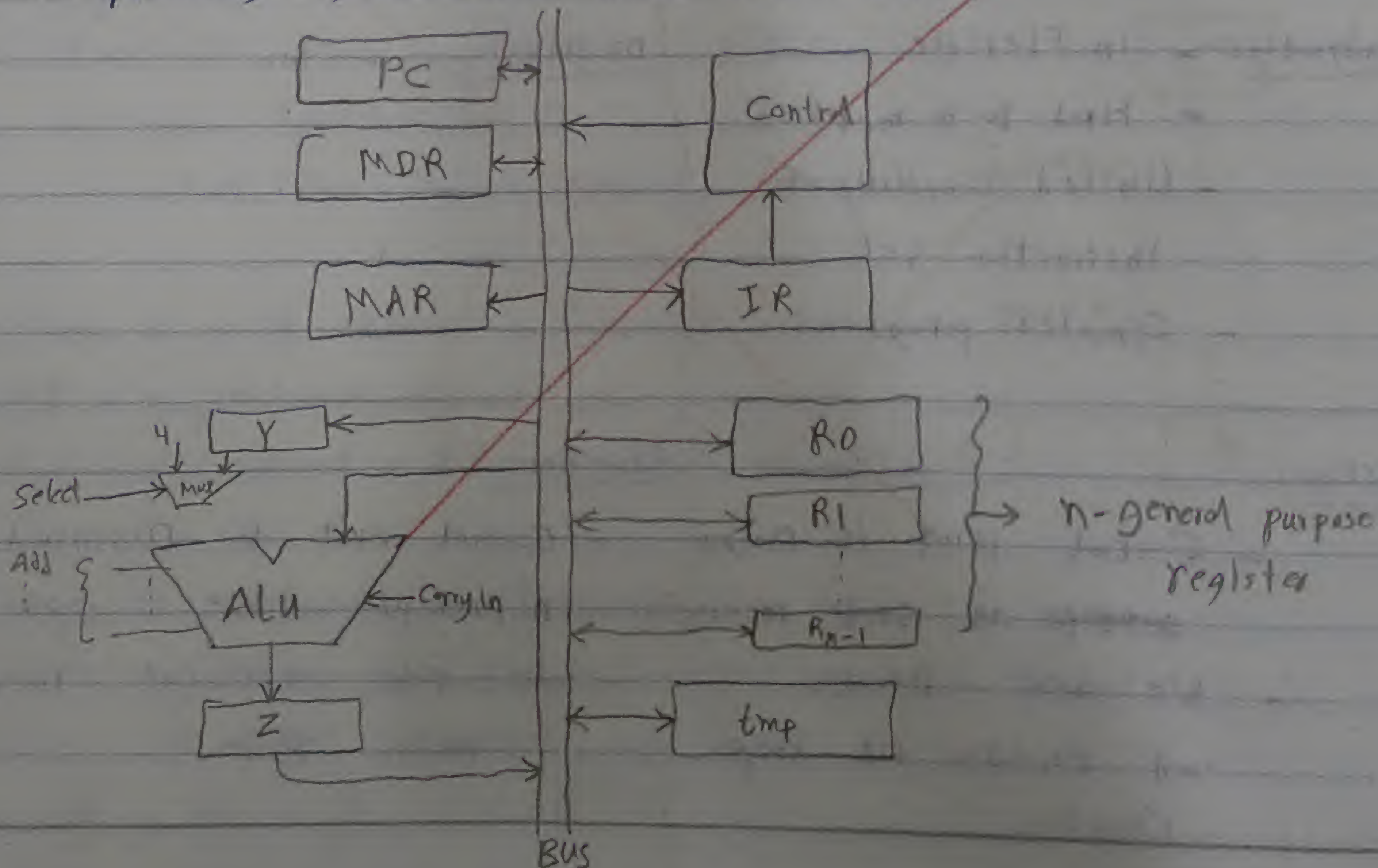
Sec 2

Question (1):

- 1) Address field contain 20 where is operand?
- Immediate
* operand is part of Instruction and has value 20
 - Direct
* operand location is the content of 20
* operand location is memory location 20
 - Indirect
* operand location is content of memory location 20
 - Register
* operand location is register 20
 - Register Indirect
* operand location is content of register 20

2) Design datapath in Single Bus organization then Add LOC, R2, R3

Assuming Instruction with two word



Add Loc, R2, R3

- 1) PC_{out}, MAR_{in}, Select 4, Add, Read, Z_{in}
- 2) ~~PC~~ Z_{out}, PC_{in}, Y_{in}, WMFC
- 3) MDR_{out}, IR_{in}
- 4) PC_{out}, MAR_{in}, Select 4, Add, Read, Z_{in}
- 5) Z_{out}, PC_{in}, Y_{in}, WMFC
- 6) MDR_{out}, MAR_{in}, Read
- 7) R2_{out}, Y_{in}, WMFC
- 8) MDR_{out}, Add, Select Y, Z_{in}
- 9) Z_{out}, R3_{in}, End

3

3

Hard wired

Advantage: Fast

Dis advantage:

- In Flexible
- Hard to be modified
- Limited complexity of instruction set
- Complex programs

Structure:

- Control Unit is Designed ~~a state~~ as State machine
- We use Decoder and Encoder and step Counter

micro programmed

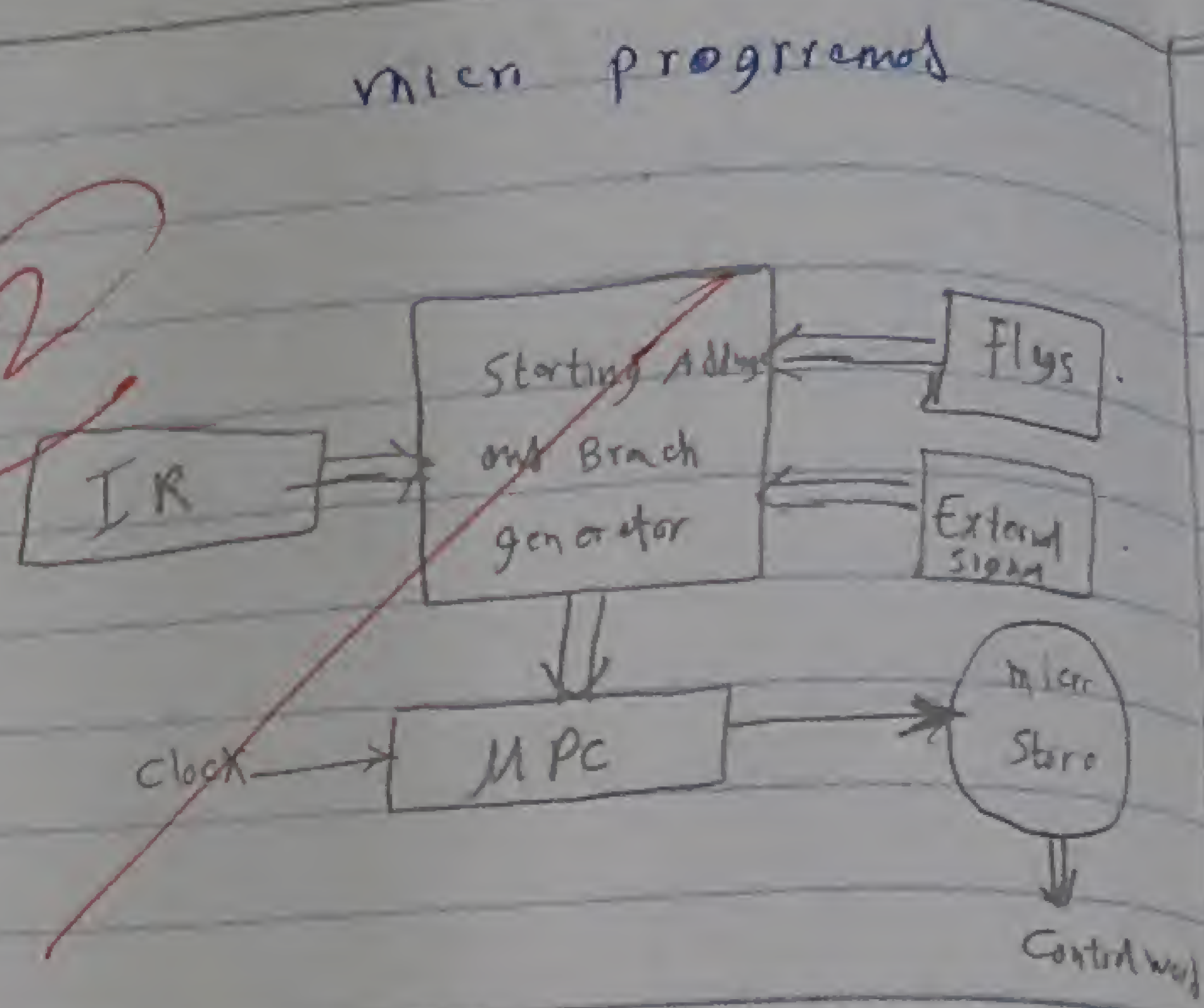
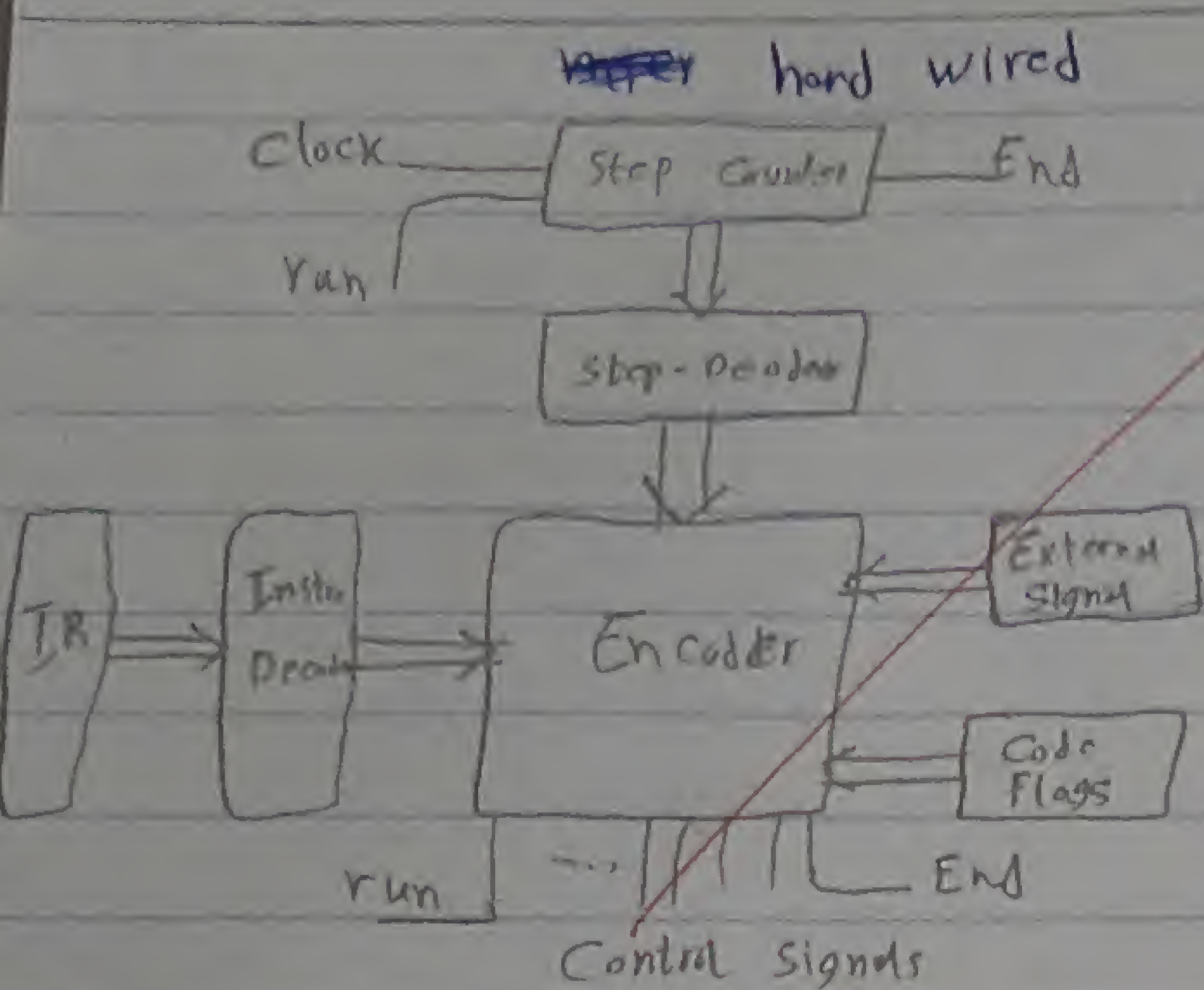
Advantage:

- Flexible
- Complex Instruction Set
- easy program

Dis Advantage: - Slow

Structure

- Control unit is Designed hierarchaly in principle like CPU Design
- We use special memory called micro Store



Question (2):

1) Difference between subroutine and Interrupt Service routine?

- 1- Subroutine perform task required by calling program
- 2- Interrupt service routine may have no common thing with the Interrupted program
- 3- Interrupt service routine and program may belong to different users
- 4- Sub routine is called by program
- 5- Interrupt service routine called by Interrupt request of I/O Device
- 6- Sub routine store only the content of PC
- 7- Interrupt service routine store PC and Condition code flags and may store registers

3) Hard w
Advantage: Fast

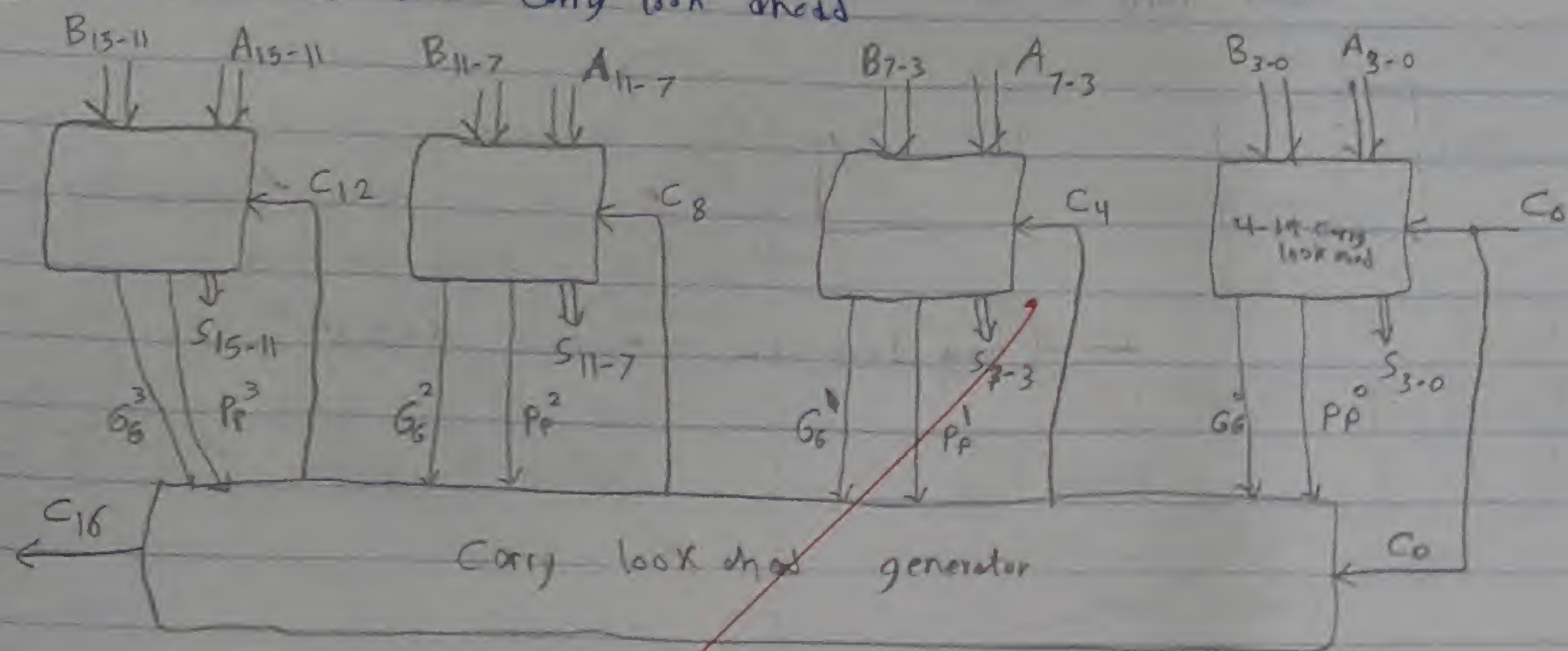
Dis advantage: -
-
- Un
- in
- Co

Structure?

- Co
- W
- Co

- 1) PC_{out}, MA
- 2) ~~PC~~ Z_{out}
- 3) MDR_{out}
- 4) PC_{out}
- 5) Z_{out}, P
- 6) MDR_{out}
- 7) RZ_{out}
- 8) MDR_{out}
- 9) Z_{out}, R

2 Design 16 bit Carry look ahead



To build this circuit we need to 4 4-bit carry look ahead

and Carry look ahead generator

$$C_{16} = G_6^3 + P_P^3 G_6^2 + P_P^3 P_P^2 G_6^1 + P_P^3 P_P^2 P_P^1 G_6^0 + P_P^3 P_P^2 P_P^1 P_P^0 C_0$$

\downarrow \downarrow \downarrow \downarrow \downarrow
 3 2 1 0 5

$C_{16} \rightarrow 5$ gates

$C_{12} \rightarrow 4$ gates

$C_8 \rightarrow 3$ gates

$C_4 \rightarrow 2$ gates

Carry look ahead need 14 gate

Inside 4-bit carry look ahead Adder Assuming that

we don't generate C_4, C_8, C_{12}, C_{16}

Inside carry look ahead need 9 gates

we have 4 4-bit carry look ahead

For each cell we need 3 gates

We have 16 Cell

Inside carry look ahead we need to generate G_i, P_i

So we need 4 gates for all G_i and 1 gate for all P_i

Total gates = $14 + 4 \times 9 + 16 \times 3 + 4 \times 4 + 1 \times 4$

= ~~118~~ gates

= 118 gates



Question (2)

3) A, B, C Connected to bus

C → high priority

A, B → Same priority (Low)

4) one Interrupt request Line

We use Interrupt Enable for that AE, BE, CE - Active High -

1- When A or B request an Interrupt

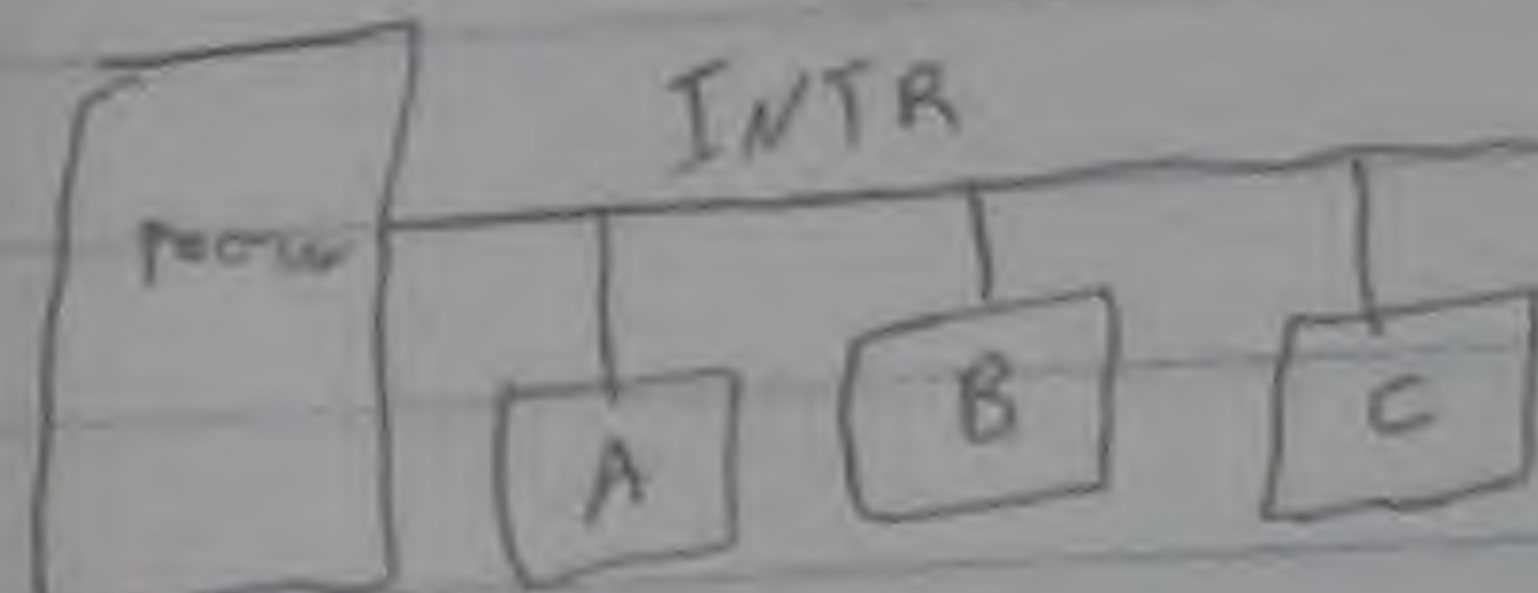
AE = 0, BE = 0, CE = 1

2- When C request an Interrupt

AE = 0, BE = 0, CE = 0

3- Default is that

AE = 1, BE = 1, CE = 1



~~We could use also Interrupt Acknowledge for that~~
~~We can~~

b) Two Interrupt Line INTR1, INTR2

we use INTR1 for C

which has higher priority

1- When A or B request an Interrupt

AE = 0, BE = 0, CE = 1

A, B will be served IF C don't request Interrupt



2- When C request an Interrupt

AE = 1, BE = 1, CE = 0 and C would be served

3- When A, B, C request an Interrupt

AE = X, BE = X, CE = 0

But only C will be served

AE = X → Don't Care